In the Claims

1.(Currently Amended) A method comprising:

receiving a uniform resource locator (URL) as associated with one of a plurality of applications requesting media content;

identifying a scheme associated with the URL;

selecting a first object operable to handle the identified scheme associated with the URL to access parameter data from a location specified by ~~a uniform resource locator (URL) based of a scheme of~~ the URL; and

based on the accessed parameter data, selecting a second object operable to read media content of a given type from the location specified by the URL ~~based on data acquired using the first object~~.

2.(Currently Amended) [[A]] The method as recited in claim 1, wherein the selection of the second object is additionally based on information contained in the URL indicating a type of the multimedia data.

3.(Currently Amended) [[A]] The method as recited in claim 1, wherein the selection of the second object is additionally based on Multipurpose Internet Mail Extensions MIME data.

4.(Currently Amended) [[A]] The method as recited in claim 1, wherein the first object is a byte stream object.

5.(Currently Amended) [[A]] The method as recited in claim 1, wherein the second object is a source object.

6.(Currently Amended) [[A]] The method as recited in claim 1, wherein the first object is a byte stream object and the second object is a source object.

7.(Currently Amended) [[A]] The method as recited in claim 1, wherein the first object is produced using a scheme handler.

8.(Currently Amended) [[A]] The method as recited in claim 1, wherein the second object is produced using a byte stream handler.

9.(Currently Amended) [[A]] The method as recited in claim 1, wherein the first object is produced by a scheme handler and the second object is produced by a byte stream handler.

10.(Currently Amended) [[A]] The method as recited in claim 1, wherein the first object is produced using a scheme handler selected from a list of two or more scheme handlers.

11.(Currently Amended) [[A]] The method as recited in claim 1, wherein the second object is produced using a byte stream handler selected from a list of two or more byte stream handlers.

12.(Currently Amended) [[A]] The method as recited in claim 1, wherein the first object is produced using a scheme handler selected from a list of two or more scheme handlers and the second object is produced using a byte stream handler selected from a list of two or more byte stream handlers.

13.(Currently Amended) [[A]] The method as recited in claim 1, further comprising accessing the multimedia data using the source object.

14.(Currently Amended) [[A]] The method as recited in claim 1, wherein the second object is produced using a byte stream handler selected from a list of byte stream handlers and wherein each byte stream handler in the list has a selection value associated therewith.

15.(Currently Amended) [[A]] The method as recited in claim 1, wherein the second object is produced using a byte stream handler selected from a list of byte stream handlers and wherein each byte stream handler in the list has a cost value associated therewith.

16.(Currently Amended) [[A]] The method as recited in claim 1, wherein the second object is produced using a byte stream handler selected from a list of byte stream handlers and wherein each byte stream handler in the list has a cost value associated therewith, the cost value indicating how many bytes must be read by the byte stream handler in determining if the byte stream handler is appropriate for selecting the second object.

17.(Currently Amended) A computer-readable medium including computer-executable instructions for performing operations comprising:

receiving a uniform resource locator (URL) specifying a location of media content as associated with one of a plurality of applications requesting media content;

determining a scheme of [[a]] the ~~uniform resource locator (URL)~~ URL ~~specifying a location of media content~~;

using the scheme to produce a byte stream object ~~that~~ to handle the determined scheme associated with the URL to access parameter data;

using the byte stream object to generate[[s]] a byte stream from the media content; and

using at least a portion of the byte stream to produce a source object that, based on the accessed parameter data, reads ~~accesses~~ the media content of a given type, from the location specified by the URL.

18.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of using the byte stream additionally includes using a file extension indicated in the URL to select the source object.

19.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a byte stream object includes choosing a scheme handler and using the chosen scheme handler to produce the byte stream object.

20.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a byte stream object includes choosing a scheme handler from a list of scheme handlers and using the chosen scheme handler to produce the byte stream object.

21.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes choosing a byte stream handler and using the chosen byte stream handler to produce the source object.

22.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes choosing a byte stream handler from a list of byte stream handlers and using the chosen byte stream handler to produce the source object.

23.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes choosing a byte stream handler from a list of byte stream handlers and using the chosen byte stream handler to produce the source object and wherein the list of byte stream handlers is ordered based on a selection values associated with the byte stream handlers.

24.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes choosing a byte stream handler from a list of byte stream handlers and using the chosen byte stream handler to produce the source object and wherein the list of byte stream handlers is ordered based on a cost values associated with the byte stream handlers.

25.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes choosing a byte stream handler from a list of byte stream handlers and using the chosen byte stream handler to produce the source object and wherein and each byte stream handler in the list has a cost value associated therewith, the cost value indicating an amount of data that must be read by the byte stream handler in determining if the byte stream handler is appropriate for producing the source object.

26.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes using a look-up process to select a subset of byte stream handlers from a set of byte stream handlers and using one of the subset of byte stream handlers to produce the source object.

27.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes using a look-up process to:

select a number of byte stream handlers; and

invoke the selected byte stream handlers one at a time until a byte stream handler produces a source object.

28.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes using a look-up process to:

select a number of byte stream handlers; and

invoke the selected byte stream handlers one at a time in a predetermined order until a byte stream handler produces a source object.

29.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes using a look-up process to:

select a number of byte stream handlers; and

invoke the byte stream handlers one at a time in a predetermined order based on cost values associated with the selected byte stream handlers until a byte stream handler produces a source object.

30.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes using a look-up process to:

select a number of byte stream handlers;

compiling a first list of byte stream handlers, each of the byte stream handlers in the first list being associated with the type of the media content;

compiling a second list of byte stream handlers, each of the byte stream handlers in the second list not being associated with the type of the media content;

invoke the byte stream handlers in the first list one at a time until either a byte stream handler in the first list produces a source object or until all byte stream handlers in the first list have been invoked without producing a source object; and

if all byte stream handlers in the first list have been invoked and none of the invoked byte stream handler from the first list produced a source object, invoking each of the byte stream handlers in the second list one at a time until either a either a byte stream handler from the second list produces a source object or until all byte stream handlers in the second list have invoked without producing a source object.

31.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes using a look-up process to:

select a number of byte stream handlers;

compiling a first list of byte stream handlers, each of the byte stream handlers in the first list being associated with the type of the media content, the byte stream handlers in the first list being ordered according to cost values associated with the byte stream handlers in the first list;

compiling a second list of byte stream handlers, each of the byte stream handlers in the second list not being associated with the type of the media content;

invoke the byte stream handlers in the first list one at a time in order until either a byte stream handler in the first list produces a source object or until all byte stream handlers in the first list have been invoked without producing a source object; and

if all byte stream handlers in the first list have been invoked and none of the invoked byte stream handler from the first list produced a source object, invoking each of the byte stream handlers in the second list one at a time until either a either a byte stream handler from the second list produces a source object or until all byte stream handlers in the second list have invoked without producing a source object.

32.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a byte stream object includes using a look-up process to:

select a number of scheme handlers; and

invoke the scheme handlers in the list one at a time until a scheme handler produces a byte stream object.

33.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a byte stream object includes using a look-up process to:

select a number of scheme handlers; and

invoke the scheme handlers in the list one at a time in a predetermined order until a scheme handler produces a byte stream object.

34.(Currently Amended) [[A]] The computer-readable medium as recited in claim 17, wherein the operation of producing a source object includes using a look-up process to:

select a number of scheme handlers;

compiling a list of scheme handlers, each of the scheme handlers in the list of scheme handlers being associated with the scheme of the URL;

invoke the scheme handlers in the list of scheme handlers one at a time until either a byte stream object is produced, a source object is produced, or all scheme handlers in the list of scheme handlers have been invoked and neither a byte stream object nor a source object have been produced;

if either a source object or a byte stream object has been produced, determining if an application has requested a source object;

if an application has requested a source object, returning the source object to the application; and

if the application has not requested a source object, compiling a list of byte stream handlers, and invoking the byte stream handlers in the first list one at a time until either a byte stream handler in the list produces a source object or until all byte stream handlers in the first list have been invoked without producing a source object.

35.(Currently Amended) A computerized system including:

an object selection module operable to:

receive a uniform resource locator (URL) as associated with one of a plurality of applications requesting media content;

determine a scheme of the URL ~~a uniform resource locator (URL)~~ specifying a location of media content;

use the scheme to produce a byte stream object ~~that~~ to handle the determined scheme associated with the URL to access parameter data;

use the byte stream object to generate[[s]] a byte stream from the media content; and

use a portion of the byte stream to produce a source object that ~~accesses~~ , based on the accessed parameter data, reads the media content of a given type, from the location specified by the URL.

36. (Currently Amended) [[A]] The computerized system as recited in claim 35, wherein the byte stream object is produced using a scheme handler.

37.(Currently Amended) [[A]] The computerized system as recited in claim 35, wherein the source object is produced using a byte stream handler.

38.(Currently Amended) [[A]] The computerized system as recited in claim 35, wherein the byte stream object is produced using a scheme handler that is selected from a list of scheme handlers, the list being selected based on the scheme of the URL.

39.(Currently Amended) [[A]] The computerized system as recited in claim 35, wherein the source object is produced using a byte stream handler that is selected from a list of byte stream handlers, the list being selected based on a byte stream generated from data at the location indicated by the URL and a portion of the URL.

40.(Currently Amended) [[A]] The computerized system as recited in claim 35, wherein the byte stream object is produced using a scheme handler that is selected from a list of scheme handlers, the list being selected based on the scheme of the URL and ordered based on cost values associated with each of the scheme handlers in the list.

41.(Currently Amended) [[A]] The computerized system as recited in claim 35, wherein the operation of producing a source object includes using a look-up process to:

select a number of byte stream handlers; and

invoke the selected byte stream handlers one at a time until a byte stream handler produces a source object.

42.(Currently Amended) [[A]] The computerized system as recited in claim 35, wherein the operation of producing a byte stream object includes using a look-up process to:

select a number of scheme handlers; and

invoke the scheme handlers in the list one at a time until a scheme handler produces a byte stream object.

43. (Currently Amended) A system comprising:

means for receiving a uniform resource locator (URL) as associated with one of a plurality of applications requesting media content;

means for identifying a scheme associated with the URL;

means for selecting a scheme handler based on ~~a scheme of a uniform resource locator (URL) specifying a location of media content~~ the identified scheme, the scheme handler producing a byte stream object operable to handle the identified scheme associated with the URL to ~~read~~ access parameter data from the location pointed to by the URL and produce a byte stream from the ~~read~~ accessed parameter data; and

means for selecting a byte stream handler based on the accessed parameter data ~~a type of the media content~~, the byte stream handler producing a source object operable to read the media content of a given type, from the location specified by the URL.

44.(Currently Amended) [[A]] The system as defined in claim 43, further comprising a lookup means for producing a list of scheme handlers, wherein the means for selecting the scheme handler selects the scheme handler from the list of scheme handlers.

45.(Currently Amended) [[A]] The system as defined in claim 43, further comprising a lookup means for producing a list of byte stream handlers, wherein the

means for selecting the byte stream handler selects the byte stream handler from the list of byte stream handlers.

46.(Currently Amended) [[A]] The system as defined in claim 43, wherein the means for selecting a scheme handler selects the scheme handler in response to a request from an application.

47.(Currently Amended) [[A]] The system as defined in claim 43, wherein the means for selecting a byte stream handler selects the byte stream handler in response to a request from an application.

48.(Currently Amended) [[A]] The system as defined in claim 43, wherein the source object produced by the byte stream handler is employed as component in a multi-component media processing pipeline.

49.(Currently Amended) [[A]] The system as defined in claim 43, wherein the source object produced by the byte stream handler is employed as component in a multi-component media processing pipeline in a media engine.

50.(Currently Amended) [[A]] The system as defined in claim 43, wherein the means for selecting a scheme handler selects the scheme handler in response to a request from an application and wherein the source object produced by the byte stream

handler is employed as component in a multi-component media processing pipeline in a media processing module that is an operational module in a operating system.

51.(Currently Amended) [[A]] The system as defined in claim 43, wherein the means for selecting a byte stream handler employs a lookup module.